Final Report: Fermilab Radiation Mapping UAV

Michael DiMeo, Michael Kifarkis, Arshad Narmawala, John Stanton, Sebastian Szulakiewicz and Bruce Treska
University of Illinois at Chicago, Chicago, IL 60607

Abstract

Fermilab's NuMI (Neutrinos at the Main Injector) target system delivers a high-energy flux of muon-neutrinos towards a detector more than 800 km away with the goal of better understanding the origin of the universe and its evolution. The target system becomes radioactive because it is irradiated by a sub-Mega-Watt power proton beam. Radiological hazard is a critical issue for a radiological worker which limits their activities near the target system; however, they often need to quantify the radiation level of the target system manually. Thus, the objective of this project is to create a radiation mapping UAV to reduce workers' exposure to radiation while improving the accuracy of the measurements. Additionally, software will process the recorded measurements into a visual intensity map which will aid in the planning of the transportation of the NuMI horn. A quadcopter UAV fitted with a radiation detector is ideal to traverse an irradiated environment as it can be deployed in any space and requires minimal hardware integration to the test room. Due to the lack of indoor GPS signal, this UAV utilizes a series of ultrasonic beacons that release several varied-frequency pulses simultaneously in order to localize the UAV. The flight controller returns a 3D coordinate tied to the radiation data to understand precisely where radiation spikes are being emitted from the horn. The flight controller stores log data that can be processed into a map of radiation intensities. Due to the inability to consistently work with a radioactive source, this prototype uses a temperature sensor and maps the thermal intensities of an indoor space.

Contents

LIST OF FIGURES

List of Tables

I. Introduction

The project sponsor—Fermilab—is a national accelerator laboratory. It is one of 17 national labs in the United States, which are all funded by the U.S. Department of Energy. Fermilab has been conducting research and running experiments centered around phenomena on the frontier of science and physics since 1967. Some of their most notable work involves their accelerator experiments, where they collide beams of high energy particles and study the aftermath of collisions in order to gain a better understanding of the behaviors of matter and energy. Fermilab employs roughly 1,750 engineers and scientists and are based in Batavia, IL; where they operate out of a 6,800-acre site, which also doubles as a nature preserve [1].

Due to the nature of Fermilab's work, radiation is a common bi-product of the experiments performed. Therefore, The purpose of this project is to provide a more accurate and safe method to measure radiation and then process the data into an easy-to-interpret visualization of radiation levels emitted from radioactive objects. The current process of measuring radiation entails a worker physically measuring radioactive objects via a portable radiation detector on the end of a telescoping rod. The worker may audibly call out the radiation levels off of the detector's screen or mentally note peak values. This process is often rushed in order to minimize operator exposure to radiation. There are many problems with this method. Exposing a human to high levels of ionizing radiation is harmful and excessive exposure to radiation can cause a multitude of illnesses, cancer or even death. As such, minimizing or eliminating operator exposure to this radiation is of paramount importance. Moreover, the current measurement method introduces large amounts of error due to the rushed nature of the test and measurement recording process. The current method provides a shallow understanding of the radiation levels emitted from the test object. By taking the time to research and develop a better way to measure radiation, exposure to human workers can be avoided, and more radiation values can be captured accurately in order to provide a better visualization of the radioactive behavior of the tested objects.

A. Literature Survey

In the process of exploring possible solutions to meet the design objective, numerous publications and several patents have been reviewed. One of the first publications, referred to the design team by the sponsor entailed a UAV created by a Berkeley Lab research team. They had developed a multi-sensor Localization and Mapping Platform which they referred to as (LAMP). The project utilizes thallium activated cesium iodide crystals which are used as scintillators to detect radiation. For positioning, the UAV uses a combination of visual cameras and LiDAR instrumentation. The UAV features on-board computers utilizing an Intel i7 processor and a discrete GPU, which is used to power the real time 3D imaging software, generated using proprietary algorithms to locate radiation hot spots. The hardware was built onto a DJI Matrice 600 UAV and features a battery life of around one hour at full computational load [2]. The components on this UAV far exceed the budget for this project. The radiation sensor alone costs well into the thousands of dollars, and the computing hardware on the UAV is top of the line. Attempting to replicate some of the methods from this design was deemed far out of the scope of this project due to the complexity of the algorithms, budget, and timeline of the project. However, reading about the LAMP UAV was a helpful starting point for the design team, as it provided an introduction to what is possible with current technology.

In terms of patented material, the design team discovered a radiation mapping UAV design developed by a team at the University of Bristol in the United Kingdom, U.S. Patent 10473794 [3]. Their design featured a positioning system comprised of a GPS device, infrared laser sensor, LiDAR and acoustic sensors. They employed a solid-state radiation sensor that acts as a gamma ray spectroscope and used proprietary software to compile radiation data and deliver them live to a receiver unit. The radiation and positioning systems are situated on a six-rotor UAV that has programmable autonomy. Like the prior LAMP UAV, this UAV is configured to be used in an outdoor setting and cover vast areas of land in search for radiation hotspots, for military and public safety applications. The UAV for this application will be used in an indoor controlled environment and does not require the array of positional equipment used on the Bristol UAV; especially considering that technologies such as GPS do not perform well indoors or underground. Again, this patent serves as a good starting point for further research, as both LiDAR and infrared systems were thoroughly explored as possible design elements.

Determining out how to acquire accurate positioning without the use of GPS was the first issue that the design team investigated at length. Liu et. al's paper "Survey of Wireless Indoor Positioning Techniques and Systems" is the primary example of the research the team encountered about indoor positioning [4]. It describes, in detail, the variety of different techniques that are commonly used to achieve accurate positioning. The methods discussed all fall under one of three techniques: triangulation, scene analysis, and proximity. Triangulation was determined to be the most viable for this application. Scene analysis required a high level of computation which was viewed as not practical for on-boarding onto a UAV or for being efficient in terms of map generation, even in post processing. Proximity methods require the use of an expansive grid of antennas, which also was deemed unreasonable for this application. Triangulation, itself, can be broken down into two individual techniques: lateration and angulation. Angulation uses the angles of received or transmitted signals between the object and fixed reference points to determine the position of the object. Further investigation of angulation methods led the design team to believe that there was not a reliable technology for implementing it due to the accuracy constraints imposed by the project. The design team found that most of the successful examples of high accuracy IPS operate via time of arrival (TOA) techniques. These techniques function by measuring an objects distance from a fixed reference points based on the propagation time of the signal sent from the reference points to the object being tracked.

Component wise, a lot of research was done on radiation sensors. The most important part of the UAV payload is the radiation detecting hardware. The heavier that hardware is, the larger and more expensive the UAV build becomes, as a stronger and heavier frame will be needed, which requires stronger and heavier motors, which requires a larger and more powerful battery to run—easily inflating the budget of the build. Therefore, choosing a lightweight and compact radiation detector was of great importance. Extensive research yielded various types of commercial radiation detecting hardware. Firstly, there are Personal Radiation Detectors, otherwise known as PRDs. These devices are small, accurate, and have fast response times. They measure many types of radiation, including alpha, beta, neutron, and gamma rays. Many kinds of technologies are used in the actual radiation detection process in these sensors, but in general, most use scintillation crystals, like the ones employed by the LAMP UAV project. Although these detectors are good quality, they are expensive and still tend to be relatively heavy in the context of a UAV build [5].

Scintillation technology relies on high energy radiation to pass through a crystalline structure. As the radiation passes through, it excites electrons in the crystalline molecules which then immediately decay, releasing a scintillating photon in the visible range. The nature of this photon allows the detector to determine the energy of the radiation and record a measurement. Geiger-Mueller (GM) technology employs tubes which are filled with gas that becomes ionized by high energy radiation. The outer casing of the tube acts as a cathode, while the wire running through the center of the tube acts as an anode. The electrons from the ionized gaseous atoms are flung towards the anode wire while the positively charged ions are flung towards the cathode tubing, due to the voltage present in the tube. This dispersion in the positively and negatively charged particles in the tube steadily increases the voltage across the tube. These voltage pulses are recorded as counts and are used to measure levels of radiation. Generally, GM tubes require much lower voltages to operate, are much smaller in size, and tend to be more affordable than scintillation counters. However, they are not as accurate, and suffer at detecting some types of radiation such as gamma [6].

Another type of radiation detector that is applicable to this project is a dosimeter. Dosimeters measure the accumulated radiation dose and are used for applications where total dosage is more important than instantaneous dosage. Many dosimeters run off Geiger-Muller technology instead of scintillation crystals and their response time are usually slower than that of a PRD. However, with the introduction of electronic dosimeters, the line between PRDs and electronic dosimeters becomes blurred [5]. Therefore, it is helpful to take a closer look at the underlying technologies at play.

In order to measure radiation values in the range of what some of the objects at Fermilab give off—around 100,000 millirem/hour—heavy duty radiation equipment is needed. For instance, even though the GM technology is known to be the more portable and lighter option, the Teletectors that are being currently used at Fermilab to hand measure radiation values, weighs in about 3 kilograms with the batteries inserted, and this detector employs GM tube technology [7]. With new advances in scintillation technology, the sponsors agreed to investigate creating a compact and light scintillation counter that could be capable of detecting radiation within the ranges necessary for some of the highly radioactive equipment needing mapping at Fermilab. They gave the design team permission to employ low cost GM tubes as a proof of concept, so the team conducted further research on portable GM tubes which yielded microcontroller-based tubes. These are integrated with micro controllers such as Arduino and Raspberry Pi and will prove useful in streamlining data acquisition and mapping [7].

B. Design Criteria

There are several generations of NuMI horns on Fermilab's campus that have not yet been used and therefore are not radioactive. Seeing the horns in person provided the team with better clarity towards reaching the objectives. The primary objective of the project is to increase the resolution and accuracy of radiation measurements whilst mitigating human exposure to ionizing radiation. Hence, the primary objective is to create a remote operated UAV that will scan for radiation around an irradiated object and produce a two dimensional radiation intensity heat map around the radioactive source. The X and Y coordinates will be position in the room and color shading will be the radiation intensity. The meetings with the project sponsors led to the formulation of the necessary preliminary requirements of the device which are outline by a product design specification table shown in Table I below. A product design specification document is a document created during the problem definition portion of the design stage. It outlines important criteria that must be considered for the product to be successful. Various design aspects are listed, which are then scrutinized under the lens of objectives, criteria, and testing conditions.

TABLE I: Product Design Specifications (PDS)

Aspect	Objective	Criteria	Test Condition
Saftey	Minimize radiation exposure to individual scanning object for radiation	Keep exposure to radiation less than that of received when scanning a NuMI horn for radiation intensity	Measure radiation at location of operator when flying the UAV
	Prevent damage to UAV components when flying in highly irradiated area	Employ an alarm or visual light that alerts the operator to move the UAV away from the high intensity reading	Make sure alarm triggers at the programmed high-level radiation reading
	Prevent damage to UAV and nearby objects in the case that the battery runs out mid flight	Utilize the battery low alarm in the UAV remote to alert the operator that the UAV must be landed and recharged	Make sure the alarm triggers when the battery reaches its low voltage limit
Accuracy	Provide an accurate radiation position and intensity map relative to the object being scanned	Accurately represent radiation intensity to the best ability of the GM tube accuracy within 15cm of positional accuracy	Calibrate GM tube with Fermilab radiation department and verify radiation map values with a Teletector that is currently used as the radiation detection method
Efficiency	Make the UAV complete a full radiation scan on one charge	Ensure UAV can complete at least one full scan to create a radiation map of the object without recharging the battery	Perform a full scan around the irradiated object and receive data within one battery charge
	Minimize weight	Keep the UAV and radiation detecting components light for maximum flight time and ability to add payload of additional sensors if needed	Weigh the entire UAV and perform a thrust test of the motor, propeller, and battery setup to get an accurate thrust:weight ratio
Cost	Keep costs low	Minimize cost for proof of concept and use as many mass-produced parts as possible for serviceability	Keep a bill of materials to tally up cost of project and document purchasing location of every part

The most important design criterion of the UAV is for it to be manually controlled by an operator and have a real-time video camera attached for remote operation. This is due to the sponsors strong desire of minimizing and/or eliminating radiation exposure to the individual responsible for measuring radiation. This data will be used to identify hot spots and allow for the objects to be properly shielded and insulated for safe transportation. A potential concern could entail the possibility of intense radiation causing damage to the UAV and the on-board electronics. A visual or audible alarm will be implemented to alert the operator if the UAV travels into zones of dangerous radiation, allowing them time to divert the UAV. In the case that the UAV runs low on battery mid-flight, an audible alarm is programmable into the remote control at the user's discretion to alert when battery is low. This feature will be tested and used to prevent crashing the UAV in the case of an insufficient charge.

Another important criterion is to ensure that the radiation heat map reflects accurate radiation intensity readings and position relative to the object. To ensure this, the Geiger Mueller radiation detector on the UAV will be calibrated by Fermilab's radiation department. It is also important for the UAV to be efficient. This is important for its ability to complete a full radiation data scan without having to recharge the battery. The UAV build heavily focuses and emphasizes on ensuring that the correct components are selected, in order to have adequate flight time. The weight of the UAV and any components has been optimized to keep the device as lightweight as possible, again adding to its efficiency and flight time.

The house of quality (HoQ) is a process that translates the functions that the sponsor desires and correlates them to engineering metrics which help drive decision making. The HoQ gives insight to the relationships between the desired functions and the design metrics as well as the design metrics that are indirectly related. The simplest example of this would be the addition of a camera to the UAV build. Of course, this addition will increase the build cost, however, the HoQ also identifies that the camera will increase power consumption and payload weight. The result is reduced flight time of the UAV. The "roof" of the HoQ indicate potential points of conflict in the design. Continuing the example above, the increase in weight has a strong negative relationship with flight time which is represented by '-'. By assigning weights to the sponsor specifications, the overall importance of each specification is calculated. The HoQ as it pertains to this design is in the below Fig. 1.

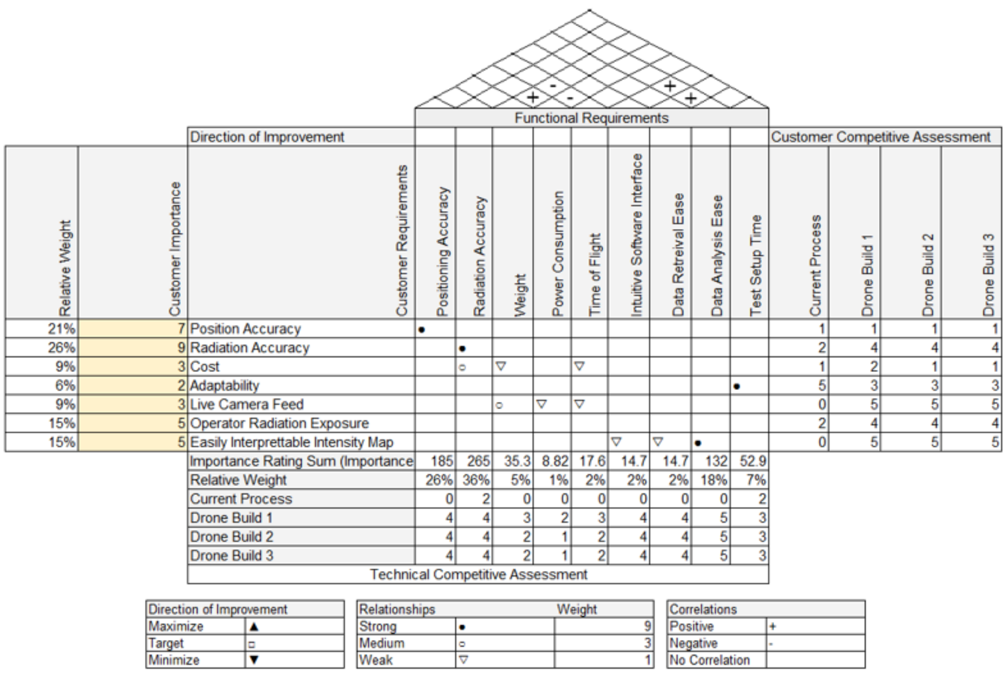


Fig. 1: The House of Quality diagram that correlates the sponsors requirement to engineering metrics.

This process highlights the relationship between increasing the robustness of the UAV and the intrinsic effect it has on the flight time. More accurate and higher range sensors have been researched; however, the equipment will drastically increase the payload weight to the UAV. Due to the inverse relationship between UAV weight and flight time, increasing the weight could cause the battery to deplete prior to test completion. For this test to be successful, flight time should be maximized as long as the data acquisition and IPS accuracy meet the acceptance criteria. The HoQ identifies position and radiation accuracy as the criteria with the highest ratings due to its high importance to the sponsor. These two criteria are essentially independent due to being strictly a hardware selection. However, time-of flight is of equal importance for successful completion of the objective despite a specific requirement not being outlined by the sponsor.

To ensure successful completion of radiation data acquisition and avoid damage to the UAV or object, a failure mode and effect analysis (FMEA) was conducted and is presented in Table II. The goal of this analysis is to identify possible design failure modes, the result of failure, the mechanism that causes failure, and the controls set to mitigate the risk. Numerical values are assigned to the failure severity, probability, and detection ability. The risk priority number (RPN) is the product of these three numerical values and indicates the importance in the design decision.

TABLE II: The Failure Mode and Effect Analysis for the S500 Quadcopter

Item/ Function	Potential Failure Mode(s)	Potential Effect(s) of Failure	Sev.	Potential Mechanism(s) of Failure	Prob.	Current Design Controls	Det	RPN	Recommended Action(s)
UAV Propeller	Break/snap	Stability loss and imminent crash	8	Impact or Collision	3	Manual Operator Controller	5	120	Replace Propeller
UAV Propeller	Break/snap	Stability loss and imminent crash	8	Degradation due to wear or radiation exposure	3	Visual Inspection	1	24	Replace Propeller
UAV Frame	Break/crack	Crash and UAV rebuild required	9	Impact or Collision	2	Visual Inspection	1	18	Replace Frame
Battery	No power	No Take off	2	Loose wiring	4	Flight qualification checklist	2	16	
Battery	Mid-air power loss	Crash	2	Loose wiring	4	Flight qualification checklist	2	16	
Battery	Low battery	Crash	3	Low battery capacity or uncharged battery	3	”Return to Launch” mode when battery is low	3	27	Connect to ArduPilot to check battery capacity
Geiger Mueller Board	Power loss	No data recorded	1	Disconnected wire or board failure	2		5	10	
Motors	No power	No take off	2	Disconnected wire, bad battery, failed PDB	2	Flight qualification checklist	2	8	
Motors	Mid-air power loss	Crash	8		2	Flight qualification checklist	2	32	

The design team has homed in on an affordable, robust, and intuitive way to achieve the sponsor’s design requirements. The selected design takes into account all of the design criteria established by the sponsor and has been evaluated based upon the metrics using processes such as decision matrices and a house of quality. The payload weight is sufficiently small to ensure a compact, efficient and cost-effective package. The small payload will allow for a flight time long enough for extensive scanning of the radioactive object. The IPS system selected has the highest positional accuracy out of all that were surveyed. Currently equipped with a CCD camera, the design allows for the operator to work remotely, reducing the risk of exposure to high levels of radiation.

II. UAV Design Overview

In order to proceed with solving the problem and have the design meet the requirements, several assumptions were made. Radiation exposure on electronics can cause permanent damage as well as data corruption. Radiation exposure can cause bit flips in digital circuits or voltage spikes in analog circuits. Furthermore, radiation creates electron-hole pairs in the electronics, changing the transistor parameters and eventually destroying them [9]. If the radioactive object is tested while it’s hot, there would be a high risk of damaging the UAV’s components by the high radiation exposure. Due to the inability to routinely test with a radioactive source, the radioactive resistance of this design is not tested. This design serves as a proof-of-concept that measurements emitted from a source can be mapped throughout a space; whether that be radiation or temperature or another measurement. Another assumption for this design is that the room that holds the radioactive objects can be prepared beforehand with positioning hardware. In addition, the radioactive object that’s being scanned will be at a static and leveled position. This will ensure that the measured radiation is accurate in regards to the position. If the crane is placed on an elevated floor, the position of the measured radiation would differ, and would not be as accurate. In similar context, the dimensions of the radioactive object relative to the room must also be known before the testing is conducted. This will minimize the hazard of crashing the UAV and makes it easier for the pilot to fly, knowing what’s surrounding the radioactive object.

A. Metrics

1) Weight: The weight of the payload that will be carried on the UAV, measured in grams. This includes all electronics, sensors, batteries, transceivers etc., which will be necessary to properly operate the UAV and perform radiation measurements. The weight of all the components that make up the payload is an important metric to track throughout the design process, as it strongly impacts design choices regarding the UAV hardware responsible for maintaining successful flight; such as the motors, propellers, chassis, speed controllers and battery. As the weight of the payload increases, so does the need for more powerful—and in many cases—more expensive hardware. The weight of the payload is currently being estimated based upon the listed weight of the components that have been selected. Once all the components have been collected, the real payload weight will be measured using a scale.

2) Positional Accuracy: Positional accuracy is defined as the distance between the true position of the UAV and the three-dimensional coordinate recorded by the indoor positioning system, measured in cm. High positional accuracy of the indoor positioning system will ensure that the UAV will capture high quality position data, with which the radiation heat map will be generated. If the position of the UAV is unable to be identified with high accuracy, then the accumulated radiation data will not be useful as it will not be tagged to the real position of measurement. This will also impact any post processing of the radiation measurements which may be done to interpolate between points and to estimate the levels of radiation at the surface of the measured object,or elsewhere in the room. Currently positional accuracy of the indoor positioning system is assumed based upon the technical specifications presented by the manufacturers of the hardware. Once the indoor positioning system has been collected, tests flights will be conducted to determine the real accuracy for necessary application. This could be done by placing markers at known distances around the test area, having the UAV fly and land on these markers, and seeing how the true position of the UAV relative to the marker compares to the coordinate measured by the indoor positioning system.

3) Radiation Detection Accuracy: The radiation detection accuracy is defined as the difference between the true value of radiation present at a given location and the radiation value recorded by the sensor, measured in mrem/hr. The accuracy of the radiation detector will determine the overall accuracy of each point on the final heat map. Like positional accuracy, this accuracy will also strongly impact any post processing of the radiation measurements which may be done to interpolate between points and to estimate the levels of radiation at the surface of the measured object or elsewhere in the room. Currently, accuracy of the radiation detector is assumed based upon the technical specifications presented by the manufacturers of the hardware. Once the detector has been collected, it will be tested and calibrated, with assistance from the sponsor, at their facility.

4) Cost: Cost is defined as the total dollars spent on the project, including all components of the UAV needed for testing and operation, measured in U.S. dollars. This cost is currently documented in the bill of materials, which includes pricing, weights and descriptions of all components currently incorporated into the design.

5) Time of Flight: Time of flight is defined as the time that the UAV can sustain flight, from take off until landing, measured in minutes and seconds. This is currently being simulated using eCalc, an online calculator which considers the weight,frame size, battery, electronic speed controller, motor, propeller, and air conditions to calculate a reasonable flight time within a $\pm 15\%$ accuracy [10]. Once the prototype UAV can fly with the actual payload, flight time will be calculated by running the UAV while collecting data until the UAV registers a low battery reading.

6) Radiation Exposure: Radiation exposure is defined as the amount of radiation the operator is exposed to during the radiation measuring process, measured in mrem/hr. The current method of measuring radiation, as indicated by the sponsor, involves a worker entering an area with levels of radiation exceeding 50,000 mrem/hr. The sponsor also indicated that the maximum limit of dosage permitted for a given worker in a year is 1,500 mrem. This project offers a method to lower the exposure of radiation to the operator of the device making the measurements. The total dosage can be measured using a dosimeter located with the operator during the entirety of data collection.

B. Proposed Solutions

1) Positioning Solutions: Determining the UAV's 3-dimensional location in the room poses a large design challenge. Due to the location and surroundings of the testing room, there is no possibility of the UAV being able to receive a GPS signal. Various indoor positioning systems (IPS) have been proposed to localize the UAV in the testing room; ultra-wideband (UWB), ultrasonic (US), optical, Wi-Fi, Bluetooth 5.1, infrared, radio frequency (RF), and light detection and ranging (LIDAR). Of these, UWB, US, and optical were deemed the most viable solutions due to meeting the preliminary criteria of accuracy, complexity, ease of use, and modularity. A decision matrix for determining the positioning system Table III demonstrates how a decision was made for the IPS selection.

With a decision matrix, the criteria that is pertinent to the design is listed and assigned a weight depending on its importance to the design. Each criterion is then given a score from 1 to 5 for each of the solutions, with 5 being the best score. Each of the criteria's definition and rating method is formally defined by Table IV.

TABLE III: Indoor Positioning System Decision Matrix

	Weighting Factor	Pozyx UWB	Marvelmind Ultrasonic Kit	Grid Mapping	QR Code
Accuracy	0.3	3	4	4	1
# of Dimensions Measured	0.05	5	5	5	3
Refresh Rate	0.1	5	2	3	5
On-Board Weight	0.25	5	5	3	5
Cost	0.05	3	4	5	5
Complexity	0.1	3	3	3	4
Scalability	0.15	4	4	5	1
Weighted Score:		**3.95**	**4**	**3.8**	**3**

TABLE IV: Criteria for the Indoor Positioning System Decision Matrix

Criteria	Definition	Rating
Accuracy	Deviations from the true position of the UAV Acceptance criteria from Fermilab: 15 cm	Below acceptance criteria: 1-2 Meets acceptance criteria: 3 Exceeds acceptance criteria: 4-5
# of Dimensions Measured	The number of dimensions the data acquires	1D: 1 2D: 3 3D: 5
Refresh Rate	Number of data points acquired per second. Refresh rate of 30 Hz is ample. Very high sample rates will likely be down sampled	$<$ 30 Hz: 1-2 30 Hz: 3 $>$30 Hz: 4-5"
On-Board Weight	Weight of the sensors/equipment that will be added to the UAV's payload. Weight of equipment not on UAV is irrelevant (anchors/wall mounted equipment/etc.)	$>$500g: 1 250g-499g: 3 $<$250g: 4-5
Cost	Cost of entire IPS/Localization equipment	$>$ \$1000: 1-2 \$500-\$999: 3 $<$ \$500: 4-5
Complexity in Design	Software Provided vs. Original Code Hardware Setup Level of Assembly/Manufacturing Required	Subjective scale: 1-5
Scalability	What needs to be changed when room is changed, or object being measured changes	Software and Hardware: 1 Software only: 3 Hardware only: 4 Nothing at all: 5

The decision matrix has identified that the Marvelmind ultrasonic method is the most appropriate solution for the given application. The ultrasonic method functions through trilateration. A frequency pulse is emitted from a beacon and the time-of-flight of the pulse is measured. Knowing the speed of sound and time-of-flight, a linear distance can be calculated. With this method, multiple beacons emit ultrasonic frequency pulses to a receiver integrated into the UAV called a "tag". The tag processes the linear distance of the beacons into a trilateration algorithm to calculate the three-dimensional position of the UAV. The ultrasonic IPS for this design uses a kit that is produced by Marvelmind Robotics [11]. One of the key benefits of the Marvelmind ultrasonic kit is the software that is provided. The open-source software simplifies the test setup process and is more reliable due to the additional filters designed into the software that processes the tag's position. The provided software reduces the complexity of the design and increases its scalability. The highest weighted criterion in the decision matrix is accuracy. This has the highest weight, as the calculated position will be tied to the acquired radiation data. If the position accuracy is low, the mapping software will identify radioactive hot spots far from the true position. Additionally, if UAV semi-autonomy is implemented in the future, low accuracy could lead to UAV collisions.

2) UAV Design Solutions: Since the application of this UAV is unique, it has been decided that a custom UAV kit will be the best option for UAV design. With a UAV, every component must be picked carefully because each component will affect the other in terms of weight as well as physical and electrical compatibility. The chassis is the first piece that must be selected. It is important that the chassis is large enough to fit not only all the components to operate the UAV, but also the accessories which in this case will be the radiation/temperature detecting equipment. There are a few key criteria which need to be considered for UAV selection. This includes thrust to weight ratio, electric motor efficiency, flight time, total weight, and cost. Thrust to weight ratio is a ratio of how much thrust the motors and propellers can generate compared to the total weight of the UAV. As a general rule of thumb, all UAV's should have at least a 2:1 thrust to weight ratio in order to be capable of taking off and having good maneuverability. Electric motor efficiency is critical since this will affect the flight time as well as motor life expectancy. Less efficient motors waste energy, creating heat, which will wear down the motors quicker as well as draw more current from the battery and shorten flight time. Flight time is another key component since the UAV needs to be able to scan the object multiple times at multiple distances. A long flight time is necessary to complete all data acquisition. Total weight will affect the load on the electric motors and require more power to sustain flight as well as reduce thrust to weight ratio. Weight needs to be kept to a minimum for an efficient and maneuverable UAV. Lastly, cost needs to be considered in order to make this solution accessible as well as having reasonable costs, should any repairs or replacements be needed to any components. The decision matrix in Table V shows the score of each UAV prototype build based on these key criteria. Each category is scored on a scale of 1-5, where 5 is the highest score. Table VI breaks down how each category of the decision matrix was scored. Table VII compares the quantitative performance calculations of each UAV build side by side.

TABLE V: UAV Decision Matrix

	Weighting Factor	S500 Quadcopter	Lynxmotion Quadcopter	S550 Hexacopter
Thrust:Weight Ratio	0.3	5	3	5
Motor Efficiency	0.2	4	3	5
Flight Time	0.25	5	3	4
Total Weight	0.15	4	5	3
Cost	0.1	4	4	4
Weighted Score:		**4.55**	**3.4**	**4.35**

TABLE VI: Criteria for the UAV Decision Matrix

Criteria	Definition	Rating
Thrust:Weight Ratio	Ratio of thrust generated to weight of UAV	<1.5: 1-2 1.5-2: 3 >2: 4-5
Motor Efficiency	Efficiency of motors at hovering operation	<80%: 1-2 80-85%: 3 >85%: 4-5
Flight Time	Length of time the UAV can sustain flight on one battery charge	<10 Minutes: 1-2 10-12 Minutes: 3 12-15 Minutes: 4 >15 Minutes: 5
Total Weight	Weight of all UAV components and accessories	>2500g: 1 2000g-2500g: 2-3 1500g-2000g: 4 1000g-1500g: 5
Cost	Total cost of the UAV	>$2000: 1 $1500-$2000: 2-3 $1000-$1500: 4 <$1000: 5

TABLE VII: UAV Performance Comparison

Criteria	S500 Quadcopter	Lynxmotion Quadcopter	S550 Hexacopter
Thrust:Weight Ratio	3.1	1.9	2.8
Motor Efficiency (%)	86.4	77.8	88.1
Hover Flight Time (min)	18.1	10	13.8
Total Weight (g)	1350	1150	2150

From Tables V and VII, it can be seen there is a close match between the S500 Quadcopter and the S550 Hexacopter builds. Both provide good thrust to weight ratio, from calculations using the Ecalc UAV performance calculator, it is estimated that the S500 will produce a 3.1:1 ratio while the S550 will produce a 2.8:1 ratio. This is due to the S500 quadcopter being able to run larger propellers since there is more spacing between the motor arms as well as being lighter since it uses less motors. Motor efficiency is close between the two. From the online calculator, the S500 quadcopter has a motor efficiency of 86.4% at hovering operation while the hexacopter has a motor efficiency of 88% at hovering operation. When it comes to flight time, this is what separates the two builds significantly. The S500 has a hover flight time of 18.1 minutes while the S550 has a hover flight time of 13.8 minutes. The S550 draws more current from the battery due to its 6 motors. Though the difference may seem small at a glance, it needs to be considered that there will be additional draw to the battery to power the radiation detecting components of the UAV and will further bring down the total flight time. Also, the flight times calculated are only at hovering operation. The UAV will draw more current rising during takeoff as well as when making maneuvers during its flight path. Therefore, having an extra 4.3 minutes will make a significant difference in being able to acquire all necessary radiation detecting data. As predicted, the S550 weighs the most due to the additional motors and ESC's. The Lynxmotion Quadcopter wins in the weight category due to having the smallest motors and least material in its chassis. This also happens to be the biggest downfall of the Lynxmotion build. Due to the smaller frame, only a 3s LiPo battery of 3500 mAh can fit in it. This significantly reduces flight time and thrust produced by its motors. This eliminates the Lynxmotion build from being pursued. Cost does fluctuate slightly between the three builds, however not by very much. The main differences between the builds are just the frames, 12 motors, and propellers. All other electronics remain the same as they have been optimized to make sure that they work together with the same protocols and give the best performance that is desired. As a result of these factors, it has been decided that the S500 Quadcopter is the best prototype to pursue due to its simplicity, flight time, size, and high thrust capability in the case of needing to add additional equipment to the UAV. All components of the proposed prototype have been verified to work with each other and a detailed list of these components can be seen in Appendix A.

C. Circuit Diagram

Due to this UAV build being completely custom, all of the wiring had to be manually soldered and various cables had to be custom made. The full diagram outlining the construction of the circuit is displayed in Fig. 7 of Appendix B. To solder the wires, a special rosin core solder along with a rosin flux had to be used in order to have good electrical connections and to avoid burning off the sheathing on the very thin gauge wires. Not all connections required soldering. The connections to the Pixhawk flight controller use JST-GH standard connector cables that comply with the Dronecode Autopilot Connector Standard.

All cables that send power are denoted by red wires, the black wires represent ground, and signal wires are blue and yellow. Port to port cable connections are represented by green wires. The remaining text in this section explains how the components of the UAV communicate with one another. All of the components on the UAV are supplied power from the four cell Lithium Polymer battery. The battery provides power directly to the power management board (PMB). The PMB then distributes power to the Pixhawk flight controller and ESC's. The ESC's are connected to the motors which spin the propellers. Since the Pixhawk is essentially the "brains" of the UAV, many of the remaining components are directly connected through its ports.

The Marvelmind hedgehog hijacks the 'GPS MODULE' port as the hedgehog is used in lieu of an actual GPS module. This is possible because the hedgehog uses the same messaging protocol as the standard Pixhawk GPS module. The radio receiver that receives the input from the operator's controller and sends the information to the Pixhawk's 'DSM/SBUS RC' port. The Pixhawk then sends the signal from 'I/O PWM Out' port to PMB's 'I/O PWM-IN' port which distributes the appropriate power to each ESC. To transmit live video, an on-screen display (OSD) module is connected to one of the 'Telemetry' ports. The other side of the OSD module diverts two ways; to the video transmitter and to the camera. The video from the camera is sent to the OSD which sends the video and telemetry data to the transmitter. The transmission is received by the operator's monitor. The final component is the radiation dosimeter; the Geiger Muller tube and board connect to the headers of the Raspberry Pi Zero which records the data logs on an SD card aboard the PI. However, the Raspberry Pi is not powered by the Pixhawk due to concerns regarding under supplied and inconsistent power delivery. To circumnavigate this, an ultimate battery eliminator circuit (UBEC) is soldered directly to the PMB into an unused ESC slot. The UBEC reduces the voltage from the battery from 12-16.8V down to 5V which is the recommended voltage for powering a Raspberry Pi.

D. Data Processing

As mentioned previously in the Circuit Diagrams section of the report, the radiation/temperature data is saved to an SD card aboard the Raspberry Pi, where it is stored as a data log. The Pixhawk records its own flight log which includes the position of the UAV during the flight. The two logs are post processed by a custom Python program which syncs the position and radiation values together by a time-based system. The program then processes the logs and outputs a two dimensional intensity map, where the XY axis are position, and the area of the map is color shaded to represent radiation/heat intensity at that location. More information regarding the intensity maps and visualizations can be read about in Software section of this report.

III. Hardware

A. Custom Brackets

Due to the large amount of components that have been added to this UAV a few custom brackets had to be made in order to be able to fit and package everything neatly and efficiently. Upon initial flight testing of the UAV, it became obvious that the original battery bracket insufficiently held the large battery we have chosen. It utilized only one battery strap with no additional support, which resulted in the battery swinging around during maneuvering. Along with that, the Marvelmind hedgehog needed a place to be mounted. The Marvelmind specifications called for the beacon to be mounted horizontally and facing downwards. Keeping this in mind, a new battery bracket was designed which utilized two battery straps. One strap was used on each end, and a center "sleeve" was added which provides extra support for the battery and doubled as a mounting base for the beacon. The bracket is comprised of two parts. The top portion was laser cut from acrylic. This section has all of the mounting holes to mount it onto the UAV frame, as well as mounting holes for the bottom portion of the bracket and slots for the battery straps. The bottom section was 3D printed and has mounting holes to attach it to the top bracket. The beacon is strapped to the bottom portion using a high strength adhesive 3M Velcro tape. This bracket is shown below in Fig. 2.

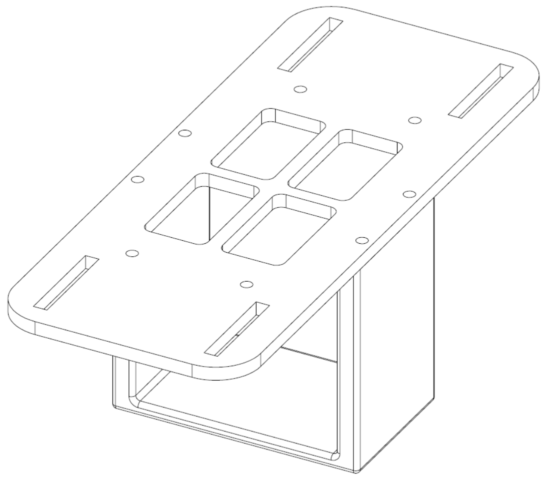
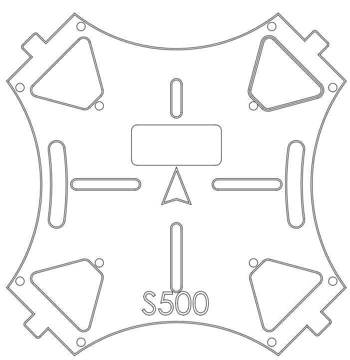
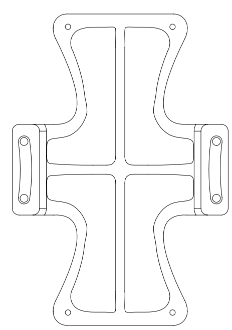


Fig. 2: Drawing of the custom battery bracket design.

The Geiger Muller (GM) radiation detector is another large part that had to be packaged onto the UAV. It is important for this part to remain as centralized as possible on the UAV for optimal weight distribution. With this in mind, it was decided that the best place to mount this accessory was on top of the upper level plate of the frame. There are two arc shaped slots on the top plate which were utilized as the locating holes of this bracket. These arcs can be seen in Fig. 3a. Two matching arc protrusions were made on the bottom of the feet of the GM bracket along with screw holes at each end to cleanly and firmly fasten the piece on to the UAV, seen in Fig. 3b. The Pixhawk flight controller was placed directly below this bracket on the top plate, so the bracket was made tall enough to allow for the various wires to be plugged in to the flight controller. The center sections of the bracket were hollowed out to save weight as well as allow access to the wiring of the flight controller. Once the bracket was printed and the UAV was tested with the GM on, an issue arose. There is a micro-USB port on the side of the flight controller which needs to be accessible in order to update, tune, and program the Pixhawk. Thus, the bracket design was modified to add an access port on the corresponding bracket leg to allow for USB port accessibility. The final bracket design can be seen below in Fig. 4.



(a)



(b)

Fig. 3: Drawings of top frame plate (a) and GM bracket (b) showing the corresponding arcs for location.

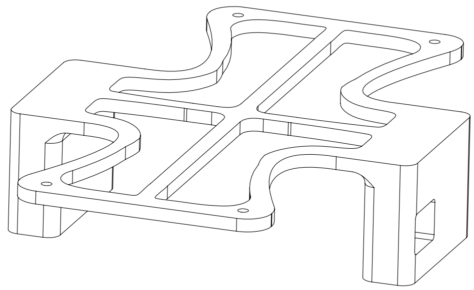


Fig. 4: Drawing of the Geiger Muller board bracket design.

The final components of the UAV design that needed a separate mounting bracket was the camera and the components that allow it to transmit live video to the monitor. The camera must be capable of facing the front of the UAV so that the pilot can have the proper perspective while flying the UAV. We are not implementing a mobile mount, such as a gimbal mount, so this meant that the mount had to be designed to fit on the front of the UAV. The lower base plate of the UAV frame, which holds the power distribution board, has three rounded slots at the front that were perfect for this application. The mount is intentionally designed to fit into the two outermost slots and screw in from the top using two to four M4 screws. The mount has a tray that is designed to hold the mini OSD board which interfaces directly with the camera itself and allows for telemetry data to be streamed directly to the video monitor on the ground. The tray is intentionally designed to fit between the brackets that attach the battery to the mounting tubes on the frame of the UAV. Additionally, there are two arms that extend from the main supports of the mount that taper down to width of the camera. The length of the arms is set to a sufficient distance as to give enough room to fit cables between the camera and the mini OSD. The arms each have a set screw hole to allow for the M2 screws to fit through and attach the camera to the mount and to fix it at a given angle for proper video capture. The final camera mount design is presented in Fig. 5.

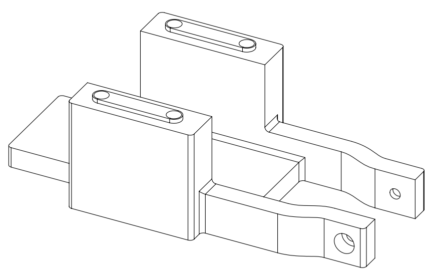


Fig. 5: Drawing of the camera mount bracket design.

B. UAV Assembly

Once UAV components started arriving, assembly immediately began. The first step in assembly was planning out where each of the main UAV components would be placed. The power management board was the first piece that had to be placed since everything on the UAV will receive its power from this component. It was decided that the board should be mounted on the lower frame plate. Once this was decided, the electronic speed controllers (ESC's) had to be soldered to the board prior to mounting. Each ESC had a power and ground wire, along with a signal and ground pair of wires which controlled the ESC operations. Once the ESC's were soldered, the board was mounted. Two of the mounting holes on the power management board lined up with two pre-existing holes on the lower plate, so screws were used to fasten that end of the board. The other end of the board was fastened using strong double-sided 3M adhesive tape. Next, the landing gear, battery bracket, arms, and the upper frame plate were added to the lower plate. The ESC's were fastened to the arms with zip-ties, and the motors were fastened to the arms with M2.5 screws that had thread-lock applied to avoid loosening under vibration. The original motor wires came very long with male bullet style connectors at the ends. The wires were trimmed down to the necessary length and the bullet connector ends were soldered to each ESC. Female bullet connectors were soldered to the shortened motor wires to allow for easy disconnecting, should a motor replacement be necessary.

The Pixhawk flight controller was mounted to the upper frame plate using the double-sided foam tape which came with it. The radio receiver was mounted to the back end of the lower frame plate using Velcro tape to allow for easy removal and addition of cables. The camera was mounted to its 3D printed bracket, the bracket was mounted at the front of the lower frame plate into the pre-existing slots. The mini OSD module was mounted with Velcro tape on its designated spot on the camera bracket. The GM tube bracket was mounted to the upper frame plate using M4 screws in the pre-existing arc slots. The UBEC which powers the Raspberry Pi and GM tube was soldered to the power management board and mounted to the bottom of the lower frame plate using double-sided adhesive tape. All of the wiring was routed and tucked away to avoid any impact with the propellers during flight. The hedgehog beacon was mounted to the bottom of the battery bracket using Velcro tape, and is powered by its own internal battery. The propellers of the UAV were mounted to the motors using the provided M2.5 screws. A full assembly of the UAV and each individual component along with its placement can be found in Fig. 10 of Appendix C.

C. *Temperature Sensor Substitute*

Due to the inability to scan radiation in any facility outside of Fermilab, the team decided to use a temperature sensor compatible with the Raspberry Pi in order to develop the scanning methods and mapping software. Once the mapping software and scanning methods are properly met with temperature data, the temperature sensor signal will be substituted with the radiation detector signal. A DS18B20 digital temperature sensor was selected due to its low cost and simple one-wire data transfer design. This was mounted in a small breadboard placed on the GM bracket and jumper wires connected from the breadboard to the respective pins on the Raspberry Pi.

IV. SOFTWARE

A. *UAV Calibration*

Once the UAV was fully assembled without the propellers installed, all of the software between the radio controller, radio receiver, ESC's, and flight controller had to be linked together. To begin, the radio controller and receiver firmware had to be updated by downloading the proper files from the manufacturer's website onto the radio itself. Once installed on the radio, the receiver was connected to the radio by a servo cable and the radio updated the files on the receiver. Once complete, the receiver was reinstalled onto the UAV, the UAV battery plugged in and powered up, and the radio was bound to the receiver. This was confirmed once the receiver's green LED was lit solid. Next, the Pixhawk flight controller had to be updated, programmed, and calibrated. The team had decided to use Mission Planner as the designated ground control software to configure and operate the flight controller due the extensive documentation available to the public regarding its operation. Ardupilot was also selected as the firmware package to be uploaded on board the flight controller due to the extensive documentation provided for the platform. To begin calibration, the controller was plugged into a laptop running Mission Planner via its micro-USB port and connected to the program. The "initial setup" function was used to automatically upload the desired Ardupilot firmware to the controller. The version currently running on the UAV is version 4.0.3. Following that, a multitude of tuning options and settings were able to be changed. Most importantly, the compass and accelerometer calibrations were complete. This was done by placing the UAV on a level surface and entering the calibration mode. The UAV was then rotated by hand in all orientations until the calibration status read "complete".

After this, the radio controller was calibrated. This step ensured that all controls were correctly set on the radio for controlling the UAV, as well as the proper ranges were set for the joystick travel. The ESC's had to be calibrated next. This was done by running the "auto calibration" function in the Mission Planner software. This calibration would be done by activating one motor at a time and setting the lowest throttle value necessary to rotate the motor. The lowest number to activate any of the motors was saved and stored. Once this was done, the flight controller was power cycled to boot into calibration mode. The radio throttle stick would be set to zero percent throttle, at which point the ESC's would give an audible tone. Then, the throttle stick was pushed to full throttle, at which point the ESC's would give off a series of beeps. Once all of the sounds stopped, the ESC's were fully calibrated. The final step before initial flight could be tested, was to set various fail-safe functions within the flight controller. A series of flight checks were enabled, such as battery voltage check, ESC connection checks, GPS safety switch, flight controller calibration and correct values being registered. At the time, we did not have a GPS module due to shipping delays. As a result, we were unable to arm the UAV due to it failing the GPS safety switch flight check.

In the Mission Planner software, we were able to turn off the GPS safety switch as a temporary bypass until we receive the GPS module. This proved, however, that the flight check fail-safes do indeed work.

B. Added Functionality

Upon initial flight, it was found that the UAV responded too quickly to the pilots inputs due to its high thrust-to-weight. The UAV was designed with lots of thrust available for carrying heavy payloads and long flight time. An unforeseen side effect of this was the fact that the operator had to be extremely cautious, skilled, and gentle with the controls in order to fly smoothly. It was very easy for the UAV to gain altitude and travel in any direction very quickly. Both Mission Planner and the radio controller were then reconfigured to alleviate this issue. The first step taken to making the UAV more user friendly was adjusting the throttle curves on the radio. All of the joystick inputs initially were set as completely linear one-to-one curves. This gives a "direct" feeling of the controls, but makes the UAV very difficult to control as only about twenty percent throttle was necessary to keep the UAV hovering. Any slight touch of a joystick would make the UAV move too quickly for being in an enclosed indoor environment. For this reason, all of the throttle inputs were changed to exponential curves in order to widen the range of the small joystick inputs. This resulted in making it so that the initial fifty percent of stick travel for each input was a very slow and gradual increase in throttle. Since there is no need for this UAV to travel at fast speeds, and only about twenty percent throttle is necessary to hover, the operator does not need to exceed that fifty percent throttle input in an indoor environment for scanning radiation.

Further steps were taken to improve the controllability via the flight controller software in Mission Planner. One tuning parameter of the flight controller is setting the value for creating the ideal exponential response curve of the motors for smoother control. The MOT_THST_EXPO tuning parameter, which is the throttle exponential value, was adjusted for the 12-inch-wide propellers being used on this UAV. Once this was done, various flight modes were tested.

The flight controller offers 24 different flight modes for various flying scenarios. A three-position switch on the radio was designated as the flight mode switch. It was decided that the three best flight modes to assign to this switch were "Stabilize", "Alt Hold", and "Land". The "Stabilize" flight mode is a simple flight mode where it allows for fully manual controls of the UAV, with stabilization activated so that when moving in a direction while flying and coming to a hover, the UAV will self-stabilize to hover level. This is the basic flight mode to default to. The "Alt Hold" flight mode is the mode which will most likely be used during radiation scanning operations. This flight mode makes for very smooth altitude climbing and descent by limiting maximum throttle as well as having a large buffer range where between 40-60% throttle input holds the UAV at a constant altitude using the built-in barometer in the flight controller. The same altitude is maintained when performing any maneuvers and can easily be adjusted by simply adding or subtracting throttle outside of the buffer zone. The "Land" flight mode will gently lower the UAV's altitude at a tunable rate in centimeters per second until it is on the ground, at which point it will automatically disarm the UAV and turn off the motors. This mode can be triggered at any time during flight or will automatically turn on when any of the current fail safes are triggered such as low battery voltage or no signal from the receiver to the radio controller. While the landing mode is engaged, the operator still has full control over roll, pitch, and yaw of the UAV so it can be placed where the operator deems safe. Once a consistent positioning method is integrated, this mode will be substituted with the "Return to Launch" function, which will take the UAV back to its launch location automatically.

C. Marvelmind Indoor Positioning System

The Marvelemind indoor positioning beacons come with their own software that is meant to initialize and track each of the beacons. The software is known as the *Marvelmind Robotics Dashboard*. The Marvelmind hardware kit comes with five beacons and a modem. Initially, the software is used to install all the relevant firmware on each piece of hardware and then can be used in a myriad of ways: to awaken or put to sleep any of the beacons, select or change the mobile beacon at any time, set datum height values for stationary beacons, alter the messaging protocol between beacons, adjust the refresh rate, and create maps and sub-maps to visualize movement relative to the stationary beacons.

Once the five beacons and modem are all setup with their firmware, the modem can be used to command and send signals to the five beacons. All five beacons need to be manually switched on and off, but once they are switched on, they enter a rest mode after 60 seconds. The modem is connected to a computer running the Dashboard software via USB. The Dashboard immediately detects the modem, allowing the modem to send signals to the five beacons to awaken them from their rest mode. Each beacon has a unique number that distinguishes it from the others, and double clicking on the device number in the bottom portion of the Dashboard software awakens each beacon. Out of the five beacons, four of them are used as stationary beacons, mounted to tripods—creating a test perimeter around an object of interest, whilst one of them is used as a mobile hedgehog that is mounted on the UAV. Any of the five beacons can be used as the hedgehog, and it can be toggled in the settings menu of said beacon.

When all five beacons are awakened, only four will populate the map. This is because the mobile beacon will not appear on the map until the user is ready to concretely set the location of the four stationary beacons by selecting ”Freeze” in the map options. Once the map is frozen, the mobile beacon appears on the map in a live-updating format. The map can be frozen at any time if adjustments need to be made to the position of any stationary beacons, or datum height values. The X, Y, and Z data that appears on Dashboard, for the hedge, is fed into Mission Planner by connecting the mobile beacon to the Pixhwak flight controller on the UAV. This information is then used in the graphing softare to create a visualizations of the pertinent locational and radiation/temperature data.

For the purposes of this project, the Dashboard software is only an intermediary step towards generating heat maps. Once all the beacons are turned on in the software, and locational data is generated by freezing the map, the user does not need to fiddle with the software any further throughout the data collection process. Once testing is complete, the beacons can all be put to sleep through the software, and then manually switched off.

D. Graphing Software

The Pixhawk flight controller records flight logs every time the UAV is armed. There is a telemetry radio attached to the telemetry port of the Pixhawk which streams live UAV data to the base station on Mission Planner as long as the UAV is powered on. These flight logs are also stored aboard an SD card inside of the flight controller. Once the flight is complete, the logs can be retrieved using Mission Planner wirelessly through the telemetry radio, or by connecting a micro-usb cable to the Pixhawk and manually downloading the logs.

The Raspberry Pi has its own SD card which stores the temperature/radiation logs. These can be downloaded physically through a micro-usb cable or by wireless tethering to the Pi from a computer using a FTP or SSH connection. The flight logs of the Pixhawk and temperature logs of the Pi both need to be downloaded for the temperature intensity map to be generated. Each of these logs have a ”time since boot” counter, and this is the time reference that is used to sync the two data logs together. This is crucial so that each position data point can be assigned a corresponding temperature value. The program uses an algorithm similar to binary search to associate a temperature value with the closest point, to the 3D coordinate.

The augmented binary search allow the program the find the value closest to the target in a sorted array in $\log(n)$ time. The algorithm divides the array into three imaginary pieces: the left half, the right half and the middle value. Comparing the middle value with the target allows the algorithm to deduce one of the three things: the middle value is the closest value, or if the closest value is in the left or right half. Repeating this step with smaller bounds allows us to get the closest value in $\log(n)$ time.

The map is generated using a custom script in Python. The logs of the Pixhawk and Raspberry Pi are put into the code, at which point the position values are taken from the Pixhawk in degrees latitude/longitude coordinates, and temperature values are taken from the Raspberry Pi log in degrees Celsius. In order for these position coordinates to have useful meaning, they had to be converted to a metric value. The degrees latitude and degrees longitude values were converted to meters from the prime meridian and meters from the equator. Once these values were obtained, the code obtains the minimum latitude and longitude values, and subtracts every position coordinate by those minimums as shown below. This yields a map which is scaled to the dimensions of the room, assuming the UAV flies around the perimeter of the room at some point in the flight.

```python
# Scale each latitude value
for latitude in latitudes:
    # Subtract the lowest latitude value
    latitude = latitude - minimum_latitude
    # Scale the latitude value to meters
    latitude = (latitude * (10000 / 90)) * 1000

# Scale each longitude value
for longitude in longitudes:
    # Subtract the lowest longitude value
    longitude = longitude - minimum_longitude
    # Scale the longitude value to meters
    longitude = (longitude * (10000 / 90)) * 1000
```

Since the UAV will never cover exactly every point of the room, there will always be some amount of unscanned area in the data points. To smooth out the data, improve accuracy, and generate a map spanning the full area around the heat source, the program uses linear interpolation methods from the python package `scipy` to calculate the temperature between all scanned points in the room.

V. Results

To achieve the proof of concept temperature map, the UAV was flown in an indoor warehouse around a pair of space heaters. The four Marvelmind stationary beacons were placed around each corner of the room, roughly in an 8m x 4m rectangle. The mobile beacon was mounted on the bottom of the UAV and provided a consistent stream of position data. Due to the current configuration of the beacons, the Marvelmind was only able to provide steady XY coordinate but, due to the lack of additional beacons, the Z coordinate was unstable and inaccurate. To mitigate this, in the tuning parameters of the Mission Planner software for the Pixhawk, it was set so that the indoor positioning system (IPS) was used for latitude and longitude data, while the Pixhawk's internal barometer was used for altitude measurements. This was crucial so that the UAV could be flown using altitude hold mode. With the unsteady altitude value of the Marvelmind, the UAV flew sporadically and was unable to hold steady altitude. With the internal barometer being used, altitude hold mode was much more steady, with only slight oscillations in height during flight. This flight mode was desired since it made the UAV easy to maneuver around the room at various heights. The UAV was flown all over the room in order to gain as many temperature points as possible to increase the map's accuracy. Once the flight was complete, the flight log and temperature log were both downloaded and ran through the Python mapping program.

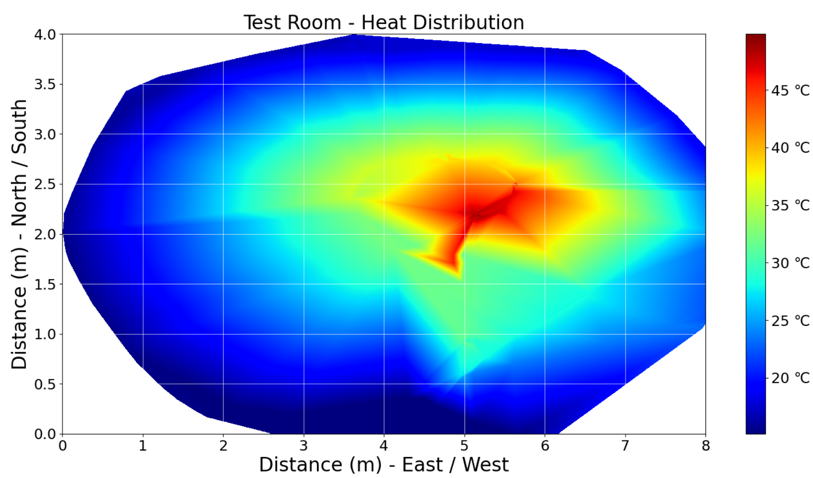


Fig. 6: Heat map generated from UAV flight using Python code.

As seen in Fig. 6, the map generated is a two dimensional heat map of the test room's temperature distribution. The X axis is the distance across the test area in the East/West directions, and the Y axis is the distance across the test area in the North/South directions. The intensity of the heat is color coded, and when the map is generated in the Python program, the user is able to hover the mouse pointer over any area of the map to get an instantaneous position coordinate and temperature reading at that exact location. The hot spot of the map is the location of the two space heaters, and the temperature was smoothly interpolated around the whole room.

Moreover, various fail-safes have been implemented to the UAV through Mission Planner. In the case of a low battery, the UAV has been programmed to softly land. The UAV battery operates at roughly 16.8V when fully charged, and the voltage slowly drops as the battery is depleted. As a cautious estimate, it has been determined that the battery is low- or at roughly 15% charge—when the voltage drops to 14V. When the Pixkhawk flight controller recognizes the voltage being below this value for more than 5 seconds, a landing sequence is initiated. At a slow and constant rate of speed, the UAV descends directly downward. During the descent, the operator is still able to control the roll, pitch, and yaw in order to find a proper landing location. This fail-safe ensures that the UAV will never fall out of the air due to a depleted battery.

Finally, the UAV demonstrates exceptional flight time at the current payload and overall weight. Currently, the UAV weighs in at approximately 1.65kg. The UAV experiences a flight time of around 22 $\pm$ 1 minutes with a fully charged battery. The UAV has a spare battery that can be conveniently swapped for another round of flight. The battery charger is able to safely charge the batteries from empty to full, in about an hour, and can charge two batteries simultaneously. In a pinch, the batteries can charge at a faster rate using a higher current, however, this is not recommended as it can compromise the longevity of the batteries.

VI. Future Work

In it's current state, the UAV is able to be flown confidently and scan a room to generate a temperature map. There are a few immediate modifications that can be made to the UAV to greatly improve its performance, flight, and scanning qualities. Currently, the Marvelmind mobile beacon is connected to the "GPS" port of the Pixhawk. This was the simplest way to substitute position data to the flight controller from the indoor positioning system. A downfall of this method is that the Marvelmind beacon does not have a compass built in to it. The Pixhawk has an internal magnetic compass which gives it the ability to stabilize the UAV and fly smoothly. The problem with this is that the flight controller is in the center of the frame, in between all of the additional electronics mounted on the UAV. These electronics give off their own magnetic fields, which cause errors in the compass and result in occasional unsteady flight. The Pixhawk's GPS module that normally is connected has a built in compass, and this module mounts on a 6 inch tall pole that moves it far away from any of the on-board electronics of the UAV. This provides an external compass to the flight controller which does not receive any distortion due to the magnetic fields of on-board electronics and yields very smooth and steady flight. The solution to this would be to migrate the mobile beacons connection to "Telemetry 2" ports of the Pixhawk and install the Pixhawk GPS into the GPS port to provide that external compass. This would require many flight controller parameter modifications and additional tuning, but would solve the occasional unsteady flight issues.

The current digital temperature sensor used is a fairly cheap and poor quality device. It has a slow refresh rate, at about one temperature reading per second, and registers temperature slowly. This requires the UAV to hover at certain areas for extended amounts of time, in order for the temperature sensor to react and update its thermal readings accordingly. This wastes operator time, and battery capacity, and makes the map development process more tedious. A higher quality digital temperature sensor can be implemented fairly easily onto the UAV with minimal change to the wiring and graphing software.

Ultimately, the temperature sensor and temperature sources were used as a proof of concept, due to the inability of obtaining a safe and robust radiation source to test readings and mapping. Therefore, the main goal is to fully replace the temperature sensor, with the micro-controller based Geiger Muller sensor. The Geiger Muller sensor was independently tested with a Raspberry Pi, and works as advertised, however, the design team was not ready to implement it onto the UAV without securing a radiation source for testing. The Geiger Muller sensor is also fairly easy to implement, as it requires only minor changes to the wiring, and the software. Although the team has opted to use microcontroller-based GM sensors, Fermilab R&D is looking into developing a compact scintillation detector that may be possibly implemented onto the UAV at a later date to truly capture the range of radiation levels emitted from objects such as the NuMI horn.

Moreover, the positional accuracy of the setup can also be vastly improved with the inclusion of three more beacons. This would provide more accurate elevation data, which would refine the accuracy of radiation maps and improve the quality of any autonomous flights. An additional beacon would be placed on the UAV, paired with the existing hedgehog, in order to obtain location and direction. Then, two further beacons would be placed on the ceiling of the room, or at a higher elevation relative to the other four stationary beacons, in order to create a vertical submap which would improve the quality of the Z value.

At a more advanced level, the team plans on exploring the possibility of streaming positional and radiation data directly to the graphing software in real time, implementing live data acquisition. Furthermore, ground station software like Mission Planner can also be used to create predetermined flight paths, implementing semi-autonomous flight. Ideally, the UAV will be programmed to fly around objects such as the NuMI horn, at predetermined elevations, and distances away from object surface. The addition of autonomous flight would increase accuracy as well as minimize risk by eliminating operator error in flight. Finally, the UAV has another 1kg worth of extra payload capacity, so further sensors and equipment can be incorporated in the future.

Appendix A
Bill of Materials

Part	Part Name	Quantity
Chassis	Readytosky S500 Quadcopter Frame Stretch X	1
ESC	Tiger Motor F35A 3-6s ESC BLHeli_32	4
Propeller	Tiger 12x4 Carbon Fiber Props	4
Motor	Tiger Motor MN3110 780KV	4
Battery	Lumenier 5200mAh 4s 35c LiPo Battery	1
Battery Straps	iFlight RC Reinforced LiPo Battery Straps 20x250mm	2
FC	Pixhawk 4	1
FC Receiver	FrSky X8R	1
Receiver Telemetry Cable	CraftandTheory Telemetry Cable for Smart Port Radios FrSky X8R	1
Controller	FrSky Taranis X9D Plus Special Edition 2019	1
Controller Battery	Hobbymate 3000mAh Lipo Battery for Frsky Taranis X9D Plus Radio Transmitter	1
IPS	Marvelmind Starter Set Super-NIA-3D	1
GPS	Holybro Pixhawk 4 Neo-M8N GPS	1
Telemetry Radio	Holybro 500mW Telemetry Radio Set (V3 for Pixhawk4) (915MHz)	1
GMTube	SBM-20	1
GMBoard	A Raspberry Pi Zero Geiger Counter / Radiation Monitor	1
TempSensor	DS18B20 Digital Temperature Sensor	1
RaspberryPI	Raspberry Pi Zero WH	1
ECCMemCard	SanDisk Extreme PRO 32GB UHS-I/U3 SDHC Flash Memory Card	1
On board camera	Foxeer Razer Micro 1200TVL 4:3 PAL/NSTC CMOS FPV Camera	1
Camera transmitter	RDQ Mach 3 Video Transmitter 25-1000mW VTX 2-6S	1
Camera receiver/monitor	FXT FX508 5.8G 40CH 5 Inch 800x480 16:9 Diversity FPV Monitor HD	1
Video Telemetry Module	Hobbypower Mini OSD module on screen display for APM pixhawk flight controller	1
Battery charger	iSDT D2 Smart Balance Charger	1
Voltage Regulator	2-6S Lipo UBEC	1
Tripod	Neewer Portable Aluminum Alloy Camera 2-in-1 Tripod Monopod Max. 70”	4
Screw	M4x10 Modified Truss Head Screw	6
Screw	M3x22 Socket Head Cap Screw	2
Screw	M3x10 Socket Head Cap Screw	8
Screw	M3x8 Socket Head Cap Screw	8
Washer	M3 Flat Washer	2
Screw	M3x7 Socket Head Cap Screw	16
Screw	M2.5x6 Socket Head Cap Screw	28
Screw	M2.5x6 Flat 100 Screw	4
Screw	M2.5x4 Socket Head Cap Screw	2
Screw	M2x12 Flat Head Screw	6
Screw	M2x12 Socket Head Cap Screw	8
Washer	M2 Flat Washer	2
Hex Nut	M2 Hex Nut	2
Spacer	M2x7 Nylon Spacer	6
Zip Tie	Small Plastic Zip Tie	11
Double Sided Tape	3M Extreme Mounting Tape Roll	1
Velcro Tape	3M Scotch Extreme Fasteners	1

Appendix B
Circuit Diagram

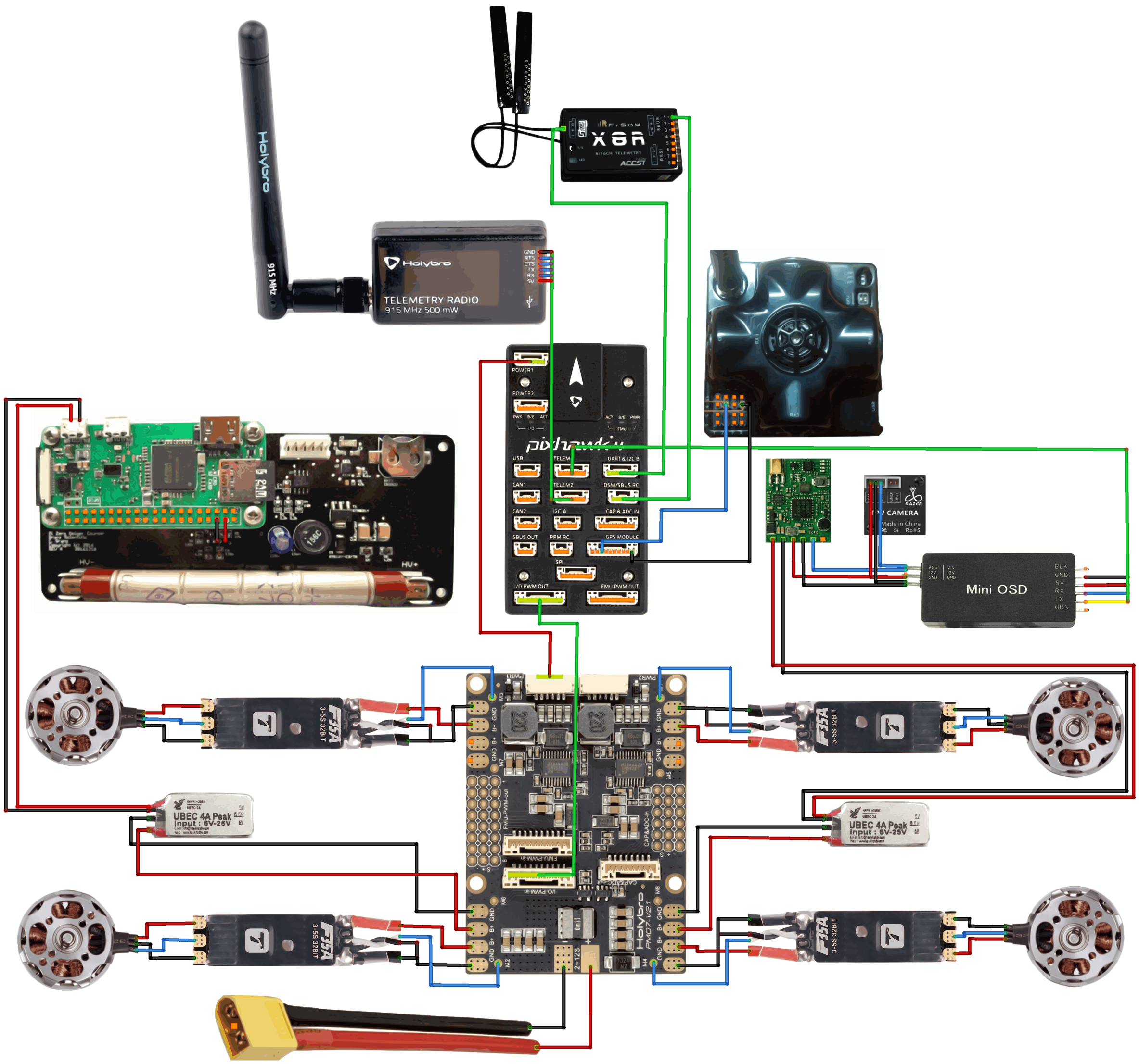


Fig. 7: Circuit diagram illustrating connections between all electronic components mounted to the UAV.

Appendix C
CAD Renderings

Fig. 8: Full color rendering of UAV assembly.

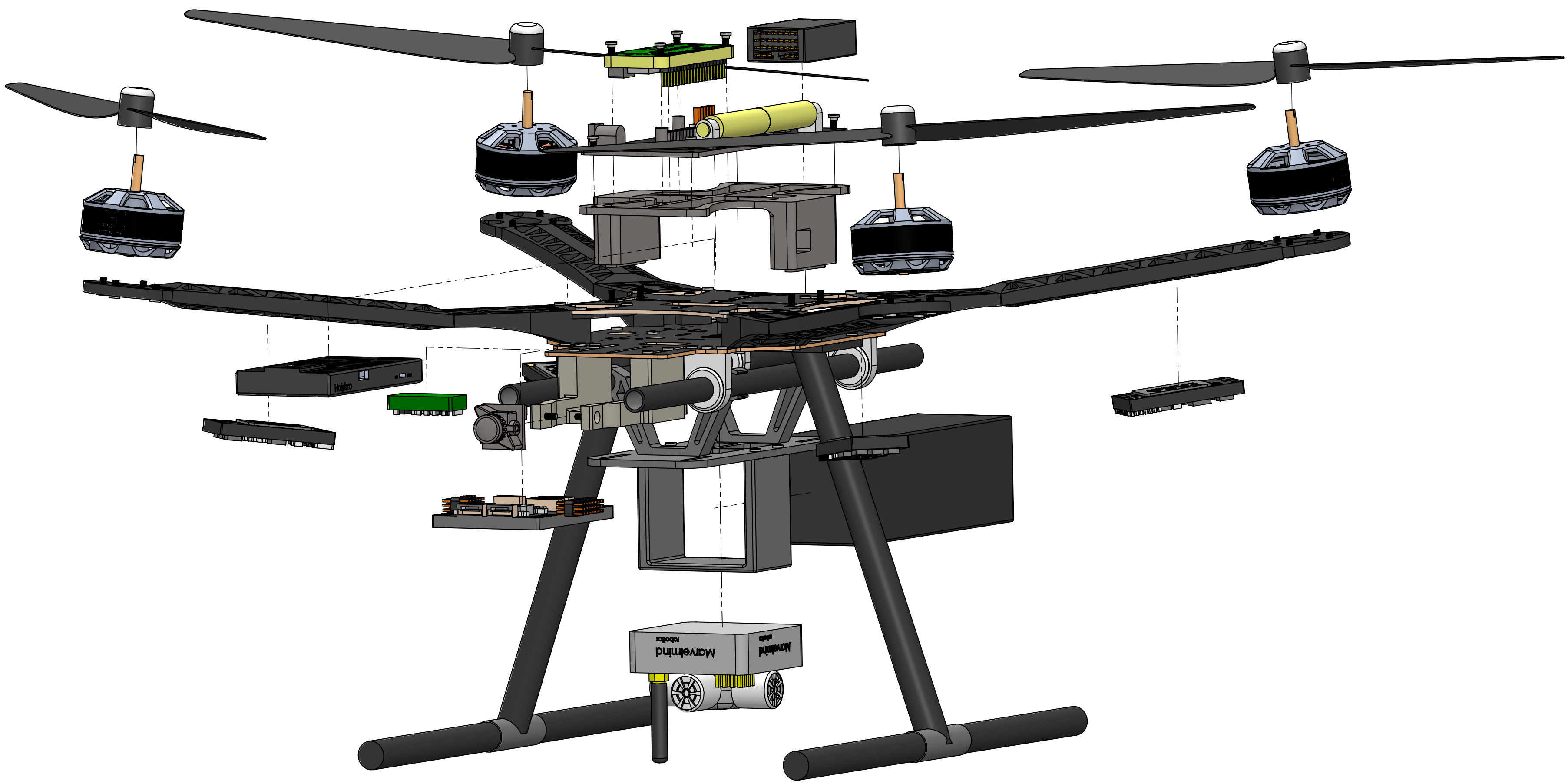


Fig. 9: Exploded view of complete UAV assembly.

ITEM NO.	PART NUMBER	QTY.
1	Frame_Bracket	4
2	Frame_Bridge	2
3	Frame_Leg	2
4	Frame_Rod	2
5	FrameBase_Bottom	1
6	Arm	4
7	FlameWheel_F450_Base_Short_Screw	14
8	FlameWheel_F450_Base_Screw	64
9	DC Motor Assembly	4
10	Propeller	4
11	PDB	1
12	FrameBase_Top	1
13	Battery_Bracket	1
14	Pixhawk	1
15	GM_Bracket	1
16	FlameWheel_F450_Base_Thin_Screw	2
17	GM_Tube	1
18	Battery	1
19	RaspberryPi	1
20	1200TVL Foxeer Micro Razer FPV Camera PAL NTSC Switchable 1.8mm lens 4ms Latency	1
21	CameraMount	1
22	FRSKY_X8R	1
23	ESC	4
24	Marvelmind Mobile Beacon	1
25	UBEC	1

Fig. 10: Complete assembly in exploded view with labels.

```python
# -----------------------------------
# ? UAV Radiation UAV Mapper
# * UIC Expo
# * Written by: Arshad Narmawala
# -----------------------------------

# * import required packages
import sys as sys
import numpy as np
import matplotlib as mp
import matplotlib.colors as colors
import matplotlib.pyplot as plt
import matplotlib.patches as mpatches
import math as math

# * import modules
from scipy.interpolate import griddata
from mpl_toolkits.mplot3d import Axes3D


def findClosest(data, start, end, target, getTarget):
    # * if only one point or the start and end have crossed
    if start == end or start > end:
        return data[start]

    # * get the mid point
    mid = int((start + end) / 2)

    # * if target is found, return point
    if getTarget(data[mid]) == target:
        return data[mid]

    # * if point is larger than target than search on the left land side
    elif getTarget(data[mid]) > target:
        return findClosest(data, start, mid - 1, target, getTarget)

    # * else search on the right hand side
    else:
        return findClosest(data, mid+1, end, target, getTarget)


if __name__ == '__main__':
    # * verify that there are three arguments
    if len(sys.argv) < 3:
        # * inform the user about the correct use
        print(
            f"Invalid use of the program"
            f"  - python mapper.py <flight log> <temperature log>"
        )

        # * exit from the program with status of 1
        exit(1)

    # * load the flight log file and get data
```

```python
flight_raw_data = None
temp_data = None
try:  # * catch expections

    # * open flight logs (data is uneven)
    with open(sys.argv[1], 'r') as file:
        flight_raw_data = file.readlines()
        file.close()

    # * open temperature logs
    temp_data = np.loadtxt(sys.argv[2], delimiter=", ")
except IOError as err:
    print(err)

# * flight data extraction
flight_data = []
for row in flight_raw_data:  # * for each row
    # * split the row using a delimiter
    row = row.split(', ')

    if row[0] == "GPS":  # * we only need gps data
        # * get the time for the point and scale it into seconds
        time_s = (int(row[1]) / 1000000) - 3

        # * get the latitude and longitude and elevation
        latitude = float(row[7])
        longitude = float(row[8])
        elevation = float(row[9])

        # * get the temperature point closest by time
        temperature = findClosest(temp_data, 0, len(
            temp_data) - 1, time_s, lambda x:  x[0])[1]

        # * add point to data
        flight_data.append((
            time_s, latitude, longitude, elevation, temperature
        ))

        # print([time_s, latitude, longitude, elevation, temperature, color])

# * transpose the data into arrays
transposed = np.transpose(flight_data)

# * Get the minimum value for latitudes
m_latitudes = []
lat_min = np.min(transposed[1].astype(np.float))

# * Get the minimum value for longitudes
m_longitudes = []
long_min = np.min(transposed[2].astype(np.float))

# * Scale the coordinates to meters from minimum lat/long
for row in transposed[1]:
    row = float(row) - lat_min
```

```
        row = (row * (10000 / 90)) * 1000
        m_latitudes.append(row)
    for row in transposed[2]:
        row = float(row) - long_min
        row = row * (10000 / 90) * 1000
        m_longitudes.append(row)

    # * create the grid
    grid_x, grid_y = np.mgrid[0:5:4000j, 0:5:4000j]
    # grid_x, grid_y = np.mgrid[0:5:50j, 0:5:50j]

    # * get the interpolated values to graph
    heat_map_data = griddata(
        points=(
            m_latitudes,
            m_longitudes
        ),
        values=transposed[4].astype(np.float),
        xi=(grid_x, grid_y),
        method="linear"
    )

    # * Graph Setup
    plt.figure(figsize=(18, 9))
    plt.title('Test Room - Heat Distribution', fontsize="24")
    plt.xlabel("Distance (m) - East / West", fontsize="24")
    plt.ylabel("Distance (m) - North / South", fontsize="24")
    plt.tick_params(axis='both', which='major', labelsize=18)
    plt.grid(color='w', linestyle='-', linewidth=0.5)

    # * Display the heatmap
    plt.imshow(
        heat_map_data.T,
        aspect="auto",
        extent=(0, 8, 0, 4),
        cmap=mp.cm.jet
    )

    # * Display the color bars
    cb = plt.colorbar(
        ticks=[15, 20, 25, 30, 35, 40, 45, 50],
        format="%d °C"
    )
    for t in cb.ax.get_yticklabels():
        t.set_fontsize(18)

    # * Display the map
    # plt.show()

    # * Save the file to map.png
    plt.savefig('map.png', transparent=True)

    # * Exit gracefully
    exit(0)
```

ACKNOWLEDGEMENT

The team would like to express their deep gratitude to Jonathan Komperda, our project advisor, for his wisdom and encouragement. The team also wishes to extend their thanks to our project sponsors at Fermilab, Katsuya Yonehara and Patrick Hurh, for offering us an opportunity to learn and grow as prospective engineers. Also the team would like to thank the UIC Makerspace for providing us with much of the tools and advice that made this project possible.

REFERENCES

[1] Fermilab National Accelerator Laboratory *Fermilab "About Fermilab." Fermilab, www.fnal.gov/pub/about/index.html.*

[2] Radiation Mapping in Real-Time *Radiation Mapping in Real-Time with the Localization and Mapping Platform LAMP from Unmanned Aerial Systems and Man-Portable Configurations.arxiv.org/ftp/arxiv/papers/1901/1901.05038.pdf.*

[3] USPTO *United States Patent and Trademark Office Patent Images, pdfpiw.uspto.gov/.piw?docid=10473794amp;PageNum=7amp;IDKey=1F441C1F9DB9amp;HomeUrl=http*

[4] Liu, H., Darabi, H., Banerjee, P. and Liu, J. *Survey of Wireless Indoor Positioning Techniques and Systems*. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), vol. 37, no. 6, pp. 1067-1080, Nov. 2007.

[5] RAE System *"Radiation Detection: Selecting the Right Equipment for the Job." Radiation Detection: Selecting the Right Equipment for the Job — RAE Systems, www.raesystems.com/customer-care/resource-center/truestories/radiation-detection-selecting-rightequipment-job.. 6*

[6] Einstein, A. *Emelue, Henry Uzoma. "Easy Biology Class." Easybiologyclass, www.easybiologyclass.com/differencebetween-gm-counterand-scintillation-counter/.*

[7] Teletector *Teletector 6112M. (2019). [ebook] LAURUS Systems. Available at: http://www.laurussystems.com/products/products_pdf/LS_am_Tele6112M.pdf [Accessed 7 Dec. 2019].*

[8] GM Board *"Geiger Counter - Radiation Sensor Board for Arduino and Raspberry Pi." Cooking Hacks by Libelium, www.cookinghacks.com/documentation/tutorials/geiger-counter-radiation-sensor-board-arduino-raspberry-pitutorialmanifesto.*

[9] Radiation Proof *"Protecting Electronics from the Effects of Radiation." www.neimagazine.com, www.neimagazine.com/features/featureprotectingelectronics-from-the-effects-of-radiation-5890599/.*

[10] Ecalc *"About ECalc - Drive Calculator." ECalc, www.ecalc.ch/index.htm?statementnotaccepted.*

[11] Marvelmind Robotics *https://marvelmind.com/product/starter-set-super-nia-3d/*